

IN THE CLAIMS:

1. (currently amended) A program product for use in a computer system that executes program steps recorded in a computer-readable medium to perform a method for updating database objects to manage version control of a database configuration in a plurality of database servers in a distributed computing network ~~to administer financial products~~, the program product comprising:

a recordable media for storing a program; and

the program of computer-readable instructions executable by the computer system to perform steps including:

(a) receiving user specified database schema files for release to a list of corresponding database servers where modifications are to occur, wherein each schema file includes proposed database object structures;

(b) verifying that the schema files contain valid DDL commands;

(c) verifying that the user has proper permissions to modify the database object structures, wherein the proper permissions are controlled by a permissions file that specifies which users have permissions to release each database object structures and the database servers that the respective database object structures are released to;

(d) comparing existing database object structures to the proposed database object structures to determine if the existing database object structures need to be modified;

(e) if the existing database object structures need to be modified, then,

i) automatically generating and executing the appropriate

commands based on the comparison to modify the existing database object structures;

ii) checking that the database servers being modified are within a respective release time;

iii) automatically generating identification files to identify the database servers that are being managed and an entire set of database object structures being managed, wherein the identification files include release permission information for each database object that specify the users who are able to release changes to each database object, the database servers to which those changes are released and specified times when changes are made to each database server while access to the identification files is limited to administrators;

iv) automatically parsing the valid DDL commands in a schema file to determine the database object structure that a respective database object should have on a database server by database server basis;

v) automatically issuing commands to modify the database object structure of the respective database object on the respective database server so the database object structure matches the database object structure that is represented in the respective schema file; and

vi) verifying that the database objects were modified properly such that if any errors occur during a structure change, an original structure of the database object is restored on the respective database server by renaming the original database object by appending a respective object name to an identifier as a backup and renaming the backup back to the original respective object name when an error occurs related to such database object;

(f) automatically creating release notes, based on the identification files, wherein the release notes are generated on a periodic basis and include documentation related to modifications of the database object structures;

(g) automatically releasing permissions files; and

(h) automatically sending the release notes to a plurality of predefined addresses so that users associated with the predefined addresses view a history of where the database objects came from in order to verify reliable storage and, as necessary, retrieval of the database object structures.

2. (original) A program product as recited in Claim 1, wherein the program further performs the step of checking that each database server that has a structure modified has the modification occur during a prescribed release time.

3. (original) A program product as recited in Claim 1, wherein the program further performs the step of receiving a selection of database servers as locations where modifications are to occur.

4. (original) A program product as recited in Claim 1, wherein the program further performs the step of verifying each modification was performed successfully.

5. (original) A program product as recited in Claim 1, wherein the program further performs the steps of receiving a plurality of schema files and bundling the plurality of schema files into a module for simultaneous execution.

6. (original) A program product as recited in Claim 1, wherein the program further performs the step of verifying that the schema files meet predefined standards.

7. (original) A program product as recited in Claim 1, wherein the program further performs the step of parsing the schema files for sequential processing.

8. (original) A program product as recited in Claim 1, wherein the program further performs the step of maintaining a history of modifications to the database object structures.

9. (original) A program product as recited in Claim 1, wherein the step of verifying that the schema files contain valid DDL commands includes enforcing a rule that the DDL commands are not modifying any database object structures other than the database object structures that the schema files specify.

10. (original) A program product as recited in Claim 1, wherein the program further performs the step of accessing identification files for identifying the database servers where modifications are occurring, wherein the identification files contain release permission information for each database object structure that is to be modified.

11. (previously presented) A method of managing a structure of database objects in a plurality of database servers in a distributed computing network such that a configuration of the structure is properly updated so that data therein is reliable, the method comprising the steps of:

(a) receiving user specified database schema files for release to a list of corresponding database servers where modifications are to occur, wherein each schema file includes proposed database object structures;

(b) verifying that the schema files contain valid DDL commands;

(c) verifying that the user has proper permissions to modify the database object structures, wherein the proper permissions are controlled by a permissions file that specifies which users have permissions to release each database object structures and the database servers that the respective database object structures are released to;

(d) comparing existing database object structures to the proposed database object structures to determine if the existing database object structures need to be modified;

(e) if the existing database object structures need to be modified, then,

i) automatically generating and executing the appropriate commands based on the comparison to modify the existing database object structures;

ii) checking that the database servers being modified are within a respective release time;

iii) automatically generating identification files to identify the database servers that are being managed and an entire set of database object structures being managed, wherein the identification files include release permission information for

each database object that specify the users who are able to release changes to each database object, the database servers to which those changes are released and specified times when changes are made to each database server while access to the identification files is limited to administrators;

iv) automatically parsing the valid DDL commands in a schema file to determine the database object structure that a respective database object should have on a database server by database server basis;

v) automatically issuing commands to modify the database object structure of the respective database object on the respective database server so the database object structure matches the database object structure that is represented in the respective schema file; and

vi) verifying that the database objects were modified properly such that if any errors occur during a structure change, an original structure of the database object is restored on the respective database server by renaming the original database object by appending a respective object name to an identifier as a backup and renaming the backup back to the original respective object name when an error occurs related to such database object;

(f) automatically creating release notes, based on the identification files, wherein the release notes are generated on a periodic basis and include documentation related to modifications of the database object structures;

(g) automatically releasing permissions files; and

(h) automatically sending the release notes to a plurality of predefined addresses so that users associated with the predefined addresses view a history of where

the database objects came from in order to verify a configuration of the database object structures has been properly updated.

12. (original) A method according to Claim 11, further comprising the step of checking that each database server that has a structure modified has the modification occur during a prescribed release time.

13. (original) A method according to Claim 11, further comprising the step of receiving a selection of database servers as locations where modifications are to occur.

14. (original) A method according to Claim 11, further comprising the step of verifying each modification was performed successfully.

15. (original) A method according to Claim 11, further comprising the steps of receiving a plurality of schema files and bundling the plurality of schema files into a module for simultaneous execution.

16. (original) A method according to Claim 11, further comprising the step of verifying that the schema files meet predefined standards.

17. (original) A method according to Claim 11, further comprising the step of parsing the schema files for subsequent processing.

18. (previously presented) A computer network for managing databases comprising:
a plurality of computers, wherein at least one of the computers has a cpu
operatively connectd to memory for storing databases; and

a source control system on at least one of the computers for managing a structure of
database objects in a plurality of database servers in a distributed computing network
wherein the source control system is for:

(a) receiving user specified database schema files for release to a list of
corresponding database servers where modifications are to occur, wherein each schema
file includes proposed database object structures;

(b) verifying that the schema files contain valid DDL commands;

(c) verifying that the user has proper permissions to modify the database
object structures, wherein the proper permissions are controlled by a permissions file that
specifies which users have permissions to release each database object structures and the
database servers that the respective database object structures are released to;

(d) comparing existing database object structures to the proposed database
object structures to determine if the existing database object structures need to be
modified;

(e) if the exisging database object structures need to be modified, then,
i) automatically generating and executing the appropriate
commands based on the comparison to modify the existing database object structures;
ii) checking that the database servers being modified are within a
respective release time;

iii) automatically generating identification files to identify the

database servers that are being managed and an entire set of database object structures being managed, wherein the identification files include release permission information for each database object that specify the users who are able to release changes to each database object, the database servers to which those changes are released and specified times when changes are made to each database server while access to the identification files is limited to administrators;

iv) automatically parsing the valid DDL commands in a schema file to determine the database object structure that a respective database object should have on a database server by database server basis;

v) automatically issuing commands to modify the database object structure of the respective database object on the respective database server so the database object structure matches the database object structure that is represented in the respective schema file; and

vi) verifying that the database objects were modified properly such that if any errors occur during a structure change, an original structure of the database object is restored on the respective database server by renaming the original database object by appending a respective object name to an identifier as a backup and renaming the backup back to the original respective object name when an error occurs related to such database object;

(f) automatically creating release notes, based on the identification files, wherein the release notes are generated on a periodic basis and include documentation related to modifications of the database object structures;

(g) automatically releasing permissions files; and

(h) automatically sending the release notes to at least one predefined address so that a user associated with the predefined address views a history of where the database objects came from and verifies a reliability of storage and access to the database object.

19. (original) A computer network as recited in Claim 18, wherein the computer network further performs the step of checking that each database server that has a structure modified has the modification occur during a prescribed release time.

20. (original) A computer network as recited in Claim 18, wherein the computer network further performs the step of receiving a selection of database servers as locations where modifications are to occur.

21. (original) A computer network as recited in Claim 18, wherein the computer network further performs the step of verifying each modification was performed successfully.

22. (original) A computer network as recited in Claim 18, wherein the computer network further performs the steps of receiving a plurality of schema files and bundling the plurality of schema files into a module for simultaneous execution.

23. (original) A computer network as recited in Claim 18, wherein the computer network further performs the step of verifying that the schema files meet predefined standards.

24. (original) A computer network as recited in Claim 18, wherein the schema files for a particular database are grouped together in a directory having a name.

25. (original) A computer network as recited in Claim 24, wherein the name is used for a name of the directory.

Claims 26-29 (cancelled).

30. (previously presented) A program product as recited in Claim 1, wherein the instructions further perform the step of selectively functioning in a what-if-mode that displays generated SQL commands for a user prior to running any such generated SQL commands.

31. (previously presented) A program product as recited in Claim 1, wherein multiple permission files are related to a single database object and the permission files designate a group of servers to which permissions in the multiple permission files apply.

32. (previously presented) A program product as recited in Claim 1, wherein a single permissions file specifies different permissions for different database servers.